receiving from a remote source a request to access one of the objects in the group of objects in the process; *B. 5,7-11*

determining whether the one object is active; and *B, 5, 60-66 7, 38-64*

activating the group of objects when it is determined that the one object is inactive to facilitate the access of the object. *B, 7, 38-64.*

29.   The method of claim 28, further including the step of:

accessing the one object in the group of objects when it is determined that the one object is active.

30.   The method of claim 29, further including the step of:

accessing a second object in the group of objects by the one object.

31.   The method of claim 28, wherein the step of activating the group of objects further includes the step of:

activating the group of objects in a virtual machine.

32.   The method of claim 31, wherein the virtual machine is a Java virtual machine.

33.   The method of claim 28, wherein the step of activating the group of objects further includes the steps of:

spawning a virtual machine; and

running the group of objects on the virtual machine.

34.   The method of claim 28, wherein the data processing system includes a second group of objects, and wherein the method further includes the steps of:

receiving from a remote source a request to access one of the objects in the second group of objects;

activating the second group of objects;

activating a virtual machine for the second group of objects; and

accessing the requested object in the second group of objects.

35.   A method in a data processing system having an object in a process, the method comprising the steps of:

receiving a request from a remote source to activate the object in the process;

determining whether the requested object is active; and

activating the object in a virtual machine when it is determined that the requested object is inactive.

36.   The method of claim 35, further including the step of:

accessing the object when it is determined that the requested object is active.

37.   The method of claim 36, further including the step of:

returning a result from the requested object.

- 3 -

38.   The method of claim 35, wherein the virtual machine is a Java virtual machine.

39.   A method in a distributed system for handling a remote object call to a server computer having a process including a group of objects, the method comprising the steps of:

receiving from a remote source a request to access an object in the group of objects;

determining whether the requested object is activated;

activating the group of objects in a virtual machine in the process on the server computer when it is determined that the requested object is not activated;

accessing the requested object when it is determined that the group of objects is active, whereupon the requested object communicates with a second object in the group of objects; and

returning a result from the accessed object.

40.   A computer-readable medium containing instructions for controlling a data processing system to perform a method, the data processing system having a group of objects in a process, the method comprising the steps of:

receiving from a remote source a request to access one of the objects in the group of objects in the process;

determining whether the one object is active; and

activating the group of objects when it is determined that the one object is inactive to facilitate the access of the object.

41.   The computer-readable medium of claim 40, wherein the method further includes the step of:

accessing the one object in the group of objects when it is determined that the one object is active.

42.   The computer-readable medium of claim 40, wherein the method further includes the step of:

accessing a second object in the group of objects by the one object.

43.   The computer-readable medium of claim 40, wherein the step of activating the group of objects further includes the step of:

activating the group of objects in a virtual machine.

44.   The computer-readable medium of claim 43, wherein the virtual machine is a Java virtual machine.

45.   The computer-readable medium of claim 40, wherein activating the group of objects further includes the steps of:

spawning a virtual machine; and

running the group of objects on the virtual machine.

46.   The computer-readable medium of claim 40, wherein the method further includes the steps of:

receiving from a remote source a request to access one of the objects in the second group of objects;

activating the second group of objects;

activating a virtual machine for the second group of objects; and

accessing the requested object in the second group of objects.

47.   A computer-readable medium containing instructions for controlling a data processing system to perform a method, the data processing system having an object, the method comprising the steps of:

receiving a request from a remote source to activate the object;

determining whether the requested object is active; and

activating the object in a virtual machine when it is determined that the requested object is inactive.

48.   The computer-readable medium of claim 47, wherein the instructions further include the step of:

accessing the object when it is determined that the requested object is active.

49. The computer-readable medium of claim 48, wherein the instructions further include the step of:

returning a result from the requested object.

50. The computer readable medium of claim 47, wherein the virtual machine is a Java virtual machine.

51. A distributed system comprising:

a first computer having:

a memory containing a first process; and

a processor for executing the first process; and

a second computer having:

a memory containing a second process containing a group of objects, and an object activator that receives from the first process a request to access one of the objects in the group of objects, that determines whether the group of objects is active, and that activates the group of objects when the object activator determines that the group is inactive to facilitate access of the object; and

a processor for running the second process and the object activator.

52. A distributed system comprising:

a first computer having:

a memory containing a first process; and

a processor for executing the first process; and

a second computer having:

a memory containing a second process containing an object, a virtual machine, and an object activator that receives a request from the first process to access the object, that determines whether the object is active, and that activates the object in the virtual machine in the second process when it is determined that the requested object is inactive; and

a processor for running the second process and the virtual machine.

53. A data processing system having a group of objects in a process, comprising:

means for receiving from a remote source a request to access one of the objects in the group of objects in the process;

means for determining whether the one object is active; and

means for activating the group of objects when it is determined that the one object is inactive to facilitate the access of the object.--

REMARKS

Applicant submits this amendment in response to the Office Action mailed December 24, 1998. Applicant has canceled claims 1-27 and added new claims 28-53 to more particularly point out and distinctly claim the subject matter of Applicant's invention.

In the Office action, the Examiner rejected claims 1-27. Specifically, the Examiner rejected claims 1, 3, 4, 6, and 23-27 under 35 U.S.C. § 103(a) as being unpatentable over Bezviner

- 8 -